CLAIMS

What is claimed is:

1   1.   A method for automatically modifying behavior of a compiled application, the

2   method comprising the computer implemented steps of:

3   receiving a first set of compiled instructions for the application; and

4   modifying the first set of compiled instructions to generate a modified set of compiled

5   instructions, including the step of inserting into said first set of compiled

6   instructions a new set of one or more instructions for invoking a modification

7   routine that provides new behavior;

8   wherein the modification routine is not part of the first set of compiled instructions.

1   2.   The method of claim 1, further comprising executing a runtime module based on the

2   modified set of compiled instructions.

1   3.   The method of claim 1, wherein:

2   the method further comprises the step of parsing the first set of compiled instructions

3   to identify an entry point to a target routine that exists in the first set of

4   compiled instructions; and

5   said step of inserting includes inserting the new set of one or more instructions at a

6   location in the first set of compiled instructions that is based on where the

7   entry point to the target routine resides in the first set of instructions.

1   4.   The method of claim 3, said step of inserting further comprising inserting the new set

2   of one or more instructions at a location within the first set of instructions that causes the new

3   set of one or more instructions to be executed immediately after the target routine is invoked.

50277-1764

1   5.   The method of claim 3, wherein:

2   the new set of one or more instructions constitutes a first new set of one or more

3   instructions; and

4   the method further comprising the steps of:

5   parsing the first set of compiled instructions to identify an exit point to the

6   target routine; and

7   inserting a second new set of one or more instructions at a location in the first

8   set of compiled instructions that is based on where an exit point to the

9   target routine resides in the first set of instructions.

1   6.   The method of claim 5, wherein

2   the modification routine stores a first time value in response to being invoked through

3   the first new set of one or more instructions; and

4   the second new set of one or more instructions invokes a second modification routine

5   for storing a second time value.

1   7.   The method of claim 6, wherein the modification routine is the same as the second

2   modification routine.

1   8.   The method of claim 6, further comprising the step of measuring performance of the

2   target routine based on a comparison between the first time value and the second time value.

1   9.   The method of claim 8, wherein the second modification routine performs said step of

2   measuring performance of the target routine.

1   10.   The method of claim 1, further comprising repeating said step of inserting for every

2   routine in the first set of compiled instructions.

50277-1764

1   11.   The method of claim 10, wherein the new set of one or more instructions are inserted

2   at each location where, within said first set of instructions, resides an entry point into one of

3   the routines in the first set.

1   12.   The method of claim 11, wherein:

2   the new set of one or more instructions constitutes a first new set of one or more

3   instructions;

4   the method further comprises the step of inserting a second new set of one or more

5   instructions for every one of the routines in the first set of compiled

6   instructions; and

7   the second new set of one or more instructions are inserted at each location in the first

8   set of compiled instructions where, within said first set of instructions, resides

9   an exit point from one of the routines in the first set.

1   13.   The method of claim 12, wherein

2   the modification routine stores a first time value in response to being invoked through

3   the first new set of one or more instructions; and

4   the second new set of one or more instructions invokes a second modification routine

5   for storing a second time value.

1   14.   The method of claim 1, wherein the new set of one or more instructions are inserted at

2   a location based on where, within said first set of instructions, resides a particular instruction

3   that is associated with entering a target routine in the first set.

1   15.   The method of claim 14, wherein the one or more instructions cause one or more

2   functions provided by said target routine to be bypassed.

1   16.   The method of claim 1, wherein the new set of one or more instructions are inserted at

2   a location based on where, within said first set of instructions, resides a particular instruction

3   that is associated with exiting a target routine in the first set.

1   17.   The method of claim 1, wherein the new set of instructions includes instructions for

2   causing the modification routine to be invoked only if a certain condition is satisfied.

1   18.   The method of claim 1, wherein said step of invoking the modification routine

2   comprises invoking a routine that accumulates data about performance of the application.

1   19.   The method of claim 18, wherein the data about performance includes data indicating

2   a number of times a particular set of instructions within said first set of compiled instructions

3   is executed.

1   20.   The method of claim 1, wherein the modification routine reports data about

2   performance of the application to a process, other than the application, that is executing on a

3   network.

1   21.   The method of claim 1, wherein the first set of compiled instructions comprise Java

2   bytecode.

1   22.   A method for automatically monitoring performance of a compiled application, the

2   method comprising the computer implemented steps of:

3   receiving a first set of compiled instructions for the application; and

4   generating a modified set of compiled instructions, including:

5   inserting a first set of one or more instructions for invoking a modification

6   routine in association with entering a particular routine within the first

7   set of compiled instructions; and

50277-1764

8   inserting a second set of one or more instructions that invoke a second

9   modification routine in association with exiting the particular routine;

10   wherein the modified set of compiled instructions causes a processor to perform the

11   steps of:

12   invoking the modification routine when entering the particular routine; and

13   invoking the second modification routine when exiting the particular routine.

1   23.   The method of claim 22, further comprising executing a runtime module based on the

2   modified set of compiled instructions.

1   24.   The method of claim 22, said step of invoking the modification routine comprising

2   accumulating first data associated with the particular routine, the first data indicating at least

3   one of a number of times the particular routine is called and a start time for the particular

4   routine.

1   25.   The method of claim 24, said step of invoking the second modification routine

2   comprising accumulating second data associated with the particular routine, the second data

3   indicating at least one of a number of times the particular routine has completed, a number of

4   active calls to the particular routine that have not completed, an end time for the particular

5   routine, an execution time based on a difference between the start time and the end time, an

6   average execution time, a minimum execution time, and a maximum execution time.

1   26.   The method of claim 22, said step of invoking the second modification routine

2   comprising recording data about performance associated with the particular routine.

1   27.   The method of claim 26, wherein said step of recording data about performance is

2   performed according to a particular schedule based on a time when the second modification

3   routine is invoked.

1 28. The method of claim 22, further comprising the step of receiving first data indicating

2 the particular routine.

1 29. The method of claim 22, wherein:

2   the method further comprises receiving first data indicating a plurality of particular

3   routines in the first set; and

4   said steps of inserting a first set and inserting a second set are repeated for each

5   particular routine of the plurality of particular routines.

1 30. The method of claim 22, said step of invoking the second modification routine

2 comprises reporting data about performance of the application to a process, other than the

3 application, that is executing on a network.

1 31. The method of claim 22, wherein the application provides a service in response to a

2 request from a client process on a network.

1 32. The method of claim 22, wherein the first set of compiled instructions comprise Java

2 bytecode.

1 33. A method for automatically monitoring performance of a compiled application, the

2 method comprising the computer implemented steps of:

3   receiving a first set of compiled instructions for the application;

4   receiving first data indicating a plurality of particular routines in the first set;

5   generating a modified set of compiled instructions, including:

6   inserting, for each particular routine of the plurality of particular routines, a

7   first set of one or more instructions for invoking a modification routine

8   in association with entering the particular routine within the first set of

9   compiled instructions; and

50277-1764

10   inserting, for each particular routine of the plurality of particular routines, a

11   second set of one or more instructions that invoke a second

12   modification routine in association with exiting the particular routine;

13   and

14   executing a runtime module based on the modified set of compiled instructions to

15   cause a processor to perform the steps of:

16   invoking the modification routine when entering the particular routine for

17   accumulating first data associated with the particular routine, the first

18   data indicating at least one of a number of times the particular routine

19   is called and a start time for the particular routine; and

20   invoking the second modification routine when exiting the particular routine

21   for

22   accumulating second data associated with the particular routine, the

23   second data indicating at least one of a number of times the

24   particular routine has completed, a number of active calls to the

25   particular routine that have not completed, an end time for the

26   particular routine, an execution time based on a difference

27   between the start time and the end time, an average execution

28   time, a minimum execution time, and a maximum execution

29   time, and

30   recording data about performance based on the first data and the

31   second data according to a particular schedule based on a time

32   when the second modification routine is invoked.

1   34.   A computer-readable medium carrying one or more sequences of instructions for

2   automatically modifying behavior of a compiled application, wherein execution of the one or

3   more sequences of instructions by one or more processors causes the one or more processors

4   to perform the steps of:

5   receiving a first set of compiled instructions for the application; and

6   modifying the first set of compiled instructions to generate a modified set of compiled

7   instructions, including the step of inserting a new set of one or more

8   instructions for invoking a modification routine that provides new behavior;

9   wherein the modification routine is not part of the first set of compiled instructions.

1   35.   The computer-readable medium of claim 34, wherein execution of the one or more

2   sequences of instructions by one or more processors causes the one or more processors to

3   further perform the step of executing a runtime module based on the modified set of compiled

4   instructions.

1   36.   The computer-readable medium of claim 34, wherein:

2   execution of the one or more sequences of instructions by one or more processors

3   causes the one or more processors to further perform the step of parsing the

4   first set of compiled instructions to identify an entry point to a target routine

5   that exists in the first set of compiled instructions; and

6   said step of inserting includes inserting the new set of one or more instructions at a

7   location in the first set of compiled instructions that is based on where the

8   entry point to the target routine resides in the first set of instructions.

1   37.   The computer-readable medium of claim 36, said step of inserting further comprising

2   inserting the new set of one or more instructions at a location within the first set of

3   instructions that causes the new set of one or more instructions to be executed immediately

4   after the target routine is invoked.

1   38.   The computer-readable medium of claim 36, wherein:

2   the new set of one or more instructions constitutes a first new set of one or more

3   instructions; and

4   execution of the one or more sequences of instructions by one or more processors

5   causes the one or more processors to further perform the steps of:

6   parsing the first set of compiled instructions to identify an exit point to the

7   target routine; and

8   inserting a second new set of one or more instructions at a location in the first

9   set of compiled instructions that is based on where an exit point to the

10   target routine resides in the first set of instructions.

1   39.   The computer-readable medium of claim 38, wherein

2   the modification routine stores a first time value in response to being invoked through

3   the first new set of one or more instructions; and

4   the second new set of one or more instructions invokes a second modification routine

5   for storing a second time value.

1   40.   The computer-readable medium of claim 39, wherein the modification routine is the

2   same as the second modification routine.

50277-1764

1   41.   The computer-readable medium of claim 39, wherein execution of the one or more

2   sequences of instructions by one or more processors causes the one or more processors to

3   further perform the step of measuring performance of the target routine based on a

4   comparison between the first time value and the second time value.

1   42.   The computer-readable medium of claim 41, wherein the second modification routine

2   performs said step of measuring performance of the target routine.

1   43.   The computer-readable medium of claim 34, wherein execution of the one or more

2   sequences of instructions by one or more processors further causes the one or more

3   processors to repeat said step of inserting for every one of routines in the first set of compiled

4   instructions.

1   44.   The computer-readable medium of claim 43, wherein the new set of one or more

2   instructions are inserted at each location where, within said first set of instructions, resides an

3   entry point into one of the routines in the first set.

1   45.   The computer-readable medium of claim 44, wherein:

2   the new set of one or more instructions constitutes a first new set of one or more

3   instructions;

4   execution of the one or more sequences of instructions by one or more processors

5   further causes the one or more processors to perform the step of inserting a

6   second new set of one or more instructions for every one of the routines in the

7   first set of compiled instructions; and

8   the second new set of one or more instructions are inserted at each location in the first

9   set of compiled instructions where, within said first set of instructions, resides

10   an exit point from one of the routines in the first set.

50277-1764

1   46.   The computer-readable medium of claim 45, wherein

2   the modification routine stores a first time value in response to being invoked through

3   the first new set of one or more instructions; and

4   the second new set of one or more instructions invokes a second modification routine

5   routine for storing a second time value.

1   47.   The computer-readable medium of claim 34, wherein the new set of one or more

2   instructions are inserted at a location based on where, within said first set of instructions,

3   resides a particular instruction that is associated with entering a target routine in the first set.

1   48.   The computer-readable medium of claim 47, wherein the one or more instructions

2   cause one or more functions provided by said target routine to be bypassed.

1   49.   The computer-readable medium of claim 34, wherein the new set of one or more

2   instructions are inserted at a location based on where, within said first set of instructions,

3   resides a particular instruction that is associated with exiting a target routine in the first set.

1   50.   The computer-readable medium of claim 34, wherein the new set of instructions

2   includes instructions for causing the modification routine to be invoked only if a certain

3   condition is satisfied.

1   51.   The computer-readable medium of claim 34, wherein said step of invoking the

2   modification routine comprises invoking a routine that accumulates data about performance

3   of the application.

1   52.   The computer-readable medium of claim 51, wherein the data about performance

2   includes data indicating a number of times a particular set of instructions within said first set

3   of compiled instructions is executed.

1   53.   The computer-readable medium of claim 34, wherein the modification routine reports

2   data about performance of the application to a process, other than the application, executing

3   on a network.

1   54.   The computer-readable medium of claim 34, wherein the first set of compiled

2   instructions comprise Java bytecode.

1   55.   A computer-readable medium carrying one or more sequences of instructions for

2   automatically monitoring performance of a compiled application, wherein execution of the

3   one or more sequences of instructions by one or more processors causes the one or more

4   processors to perform the steps of:

5   receiving a first set of compiled instructions for the application; and

6   generating a modified set of compiled instructions, including:

7   inserting a first set of one or more instructions for invoking a modification

8   routine in association with entering a particular routine within the first

9   set of compiled instructions; and

10   inserting a second set of one or more instructions that invoke a second

11   modification routine in association with exiting the particular routine;

12   wherein the modified set of compiled instructions causes a processor to perform the

13   steps of:

14   invoking the modification routine when entering the particular routine; and

15   invoking the second modification routine when exiting the particular routine.

1   56.   The computer-readable medium of claim 55, wherein execution of the one or more

2   sequences of instructions by one or more processors further causes the one or more

3   processors to perform the step of executing a runtime module based on the modified set of

4   compiled instructions.

50277-1764

1 57. The computer-readable medium of claim 55, said step of invoking the modification

2 routine comprising accumulating first data associated with the particular routine, the first data

3 indicating at least one of a number of times the particular routine is called and a start time for

4 the particular routine.

1 58. The computer-readable medium of claim 57, said step of invoking the second

2 modification routine comprising accumulating second data associated with the particular

3 routine, the second data indicating at least one of a number of times the particular routine has

4 completed, a number of active calls to the particular routine that have not completed, an end

5 time for the particular routine, an execution time based on a difference between the start time

6 and the end time, an average execution time, a minimum execution time, and a maximum

7 execution time.

1 59. The computer-readable medium of claim 55, said step of invoking the second

2 modification routine comprising recording data about performance associated with the

3 particular routine.

1 60. The computer-readable medium of claim 59, wherein said step of recording data

2 about performance is performed according to a particular schedule based on a time when the

3 second modification routine is invoked.

1 61. The computer-readable medium of claim 55, wherein execution of the one or more

2 sequences of instructions by one or more processors further causes the one or more

3 processors to perform the step of receiving first data indicating the particular routine.

50277-1764

1 62. The computer-readable medium of claim 55, wherein:

2 execution of the one or more sequences of instructions by one or more processors

3 further causes the one or more processors to perform the step of receiving first

4 data indicating a plurality of particular routines in the first set; and

5 said steps of inserting a first set and inserting a second set are repeated for each

6 particular routine of the plurality of particular routines.

1 63. The computer-readable medium of claim 55, said step of invoking the second

2 modification routine comprises reporting data about performance of the application to a

3 process, other than the application, that is executing on a network.

1 64. The computer-readable medium of claim 55, wherein the application provides a

2 service in response to a request from a client process on a network.

1 65. The computer-readable medium of claim 55, wherein the first set of compiled

2 instructions comprise Java bytecode.